Inferring Path Sharing Based on Flow Level TCP Measurements

Dogu Arifler, Gustavo de Veciana, and Brian L. Evans
Department of Electrical and Computer Engineering
The University of Texas at Austin, Austin, TX 78712-0240 USA
{arifler, gustavo, bevans}@ece.utexas.edu

Abstract— **We develop methods to infer path or bottleneck sharing among TCP flow classes based on flow level measurements available from current traffic monitoring tools. Our premise is that flows that temporally overlap on congested resources will have correlated throughputs. We propose to use factor analysis to explore the correlation structure of flow class throughputs in order to hypothesize which flow classes might share congested resources. The effectiveness of this "black box" approach is studied using empirical data. We show that making such inferences based on flow level statistics is viable in practice, and can serve as an effective, novel tool for network design and configuration decisions. Our work on inferring bottleneck sharing differs significantly from previous work in that we consider flow level instead of packet level statistics, and hence may potentially influence research in that area. Possible applications of this technique include network monitoring and root cause analysis of poor performance.**

I. Introduction

Recently, the study of Internet (IP) traffic at the flow level has attracted much attention [1], [2], [3], since flow level performance (such as the time required to transfer a file) determines the quality of service perceived by end users. A commonly accepted definition of an IP *flow* is a unidirectional sequence of packets between a source and a destination endpoint identified by common IP addresses, Transmission Control Protocol (TCP) or User Datagram Protocol (UDP) port numbers, IP protocol type, type of service fields in IP headers, etc. An IP *flow class* is a collection, or aggregation, of flows having a common attribute. For example, we will refer to all flows sharing common source and destination IP address prefixes as a flow class.

A significant portion of IP traffic consists of packets from elastic flows generated by applications such as Web transfers, file transfers (FTP), and e-mail [4]. Such flows are transported via TCP which uses packet delay and loss as indicators of the available bandwidth to adjust the data transmission window at the sender. Consequently, the quality of service (delay) perceived by such flows is determined by the variable rates achieved by TCP's congestion control algorithm over the sojourn of the flow. In this work, we deal only with classes of flows mediated through TCP. Hence, we will refer to TCP flow classes as just flow classes.

State-of-the-art networking equipment is capable of generating *flow records*. A record contains the source and destination IP addresses, source and destination port numbers, start and end times, and the size (in bytes and packets) of flows traversing that network element. A given sequence of packets sharing the same attributes may be partitioned into multiple flows based on a flow timeout interval, which can range from seconds to minutes. An end of connection indication in the TCP header (TCP FIN) may also be used to determine the end of a TCP flow. Examples of technologies capable of generating flow records are NetFlow [5], Argus [6], and sFlow [7]. A major problem in flow measurement is the lack of scalability: at very high speed routers, the number of flows to be measured might easily exceed millions per hour. Therefore, at high link speeds, the flows [8] and/or the packets within a flow [9] may be sampled by the network element when generating flow records in order to keep up with the link speeds.

A. Motivation

Packet level characteristics of IP traffic are complex to analyze due to their extreme variability over a wide range of time scales [10]. Packet level IP traffic is also data intensive to collect and store for subsequent analysis. Motivated by the increasing availability of flow level measurements and the development of methods for characterizing flow level performance of elastic traffic [1], [2], [3], we consider analyzing network flow measurements for network monitoring and inference of network properties rather than studying packet level details. Although current network management software provides extensive flow measurement capabilities, tools to analyze such data are quite limited. Such tools might be very useful to network engineers.

One critical problem that has been identified by Internet service providers (ISPs) and content providers (CPs) is determining which flows might share paths or bottlenecks in the Internet. This is usually difficult without access to the complete routing information for the network. In most cases, network managers have information only about their network domain and the properties of the other domains are unknown. By inferring which flows share paths or bottlenecks using only local measurements and the attributes of flows (e.g. source/destination IP addresses), service providers might perform load balancing of traffic that share a common bottleneck onto disjoint paths. For instance, upon inferring

D. Arifler and B. L. Evans were supported by The State of Texas Advanced Technology Program under project 003658-0614-2001.

that two customers or customer bases are experiencing poor Web performance due to a bottleneck link serving them, the provider of the Web content might choose to replicate content at a second location to reduce the load on the bottleneck link. We will term our inference of path sharing based only on measurements collected at one site and the attributes of flows as a "black box" approach.

The primary question addressed in this paper is "how to infer the source/destination pairs sharing paths or bottlenecks" based on flow measurements made at a *single* network element without access to routing information inside the network. Inferences for path or bottleneck sharing from flow measurements are based on the following premise: TCP flow classes that temporally overlap long enough on the same path, or bottleneck, will tend to have positively correlated throughputs. We propose to use *factor analysis* to study the correlation structure of flow class throughputs and to infer which flow classes share paths or bottlenecks.

B. Related Work

Our work on inferring path or bottleneck sharing differs significantly from the previous work in that we consider *flow level* instead of *packet level* statistics. Furthermore, we rely on passive flow measurements made at a networking element (e.g. router, gateway, server) rather than using end-to-end probes, as has been the focus of previous work. While many other previous correlation based methods that infer path sharing are limited to determining whether particular flow class pairs are correlated, the method developed in this paper considers a set of flow classes simultaneously.

The simplest approach to detecting shared paths is to use a utility such as `traceroute` that tracks the route that a packet follows from its source to its destination. Such utilities, however, require the cooperation of routers in the network on the path of the flow. Owners of the carrier networks are often unwilling to provide information about their networks, and hence the use of such utilities is not always viable. Savage, Cardwell, and Anderson [11] describe an approximation for detecting shared paths by looking at the destination IP addresses of flows. Their approximation is based on the fact that flows destined to a particular host or network address generally follow the same path, and hence visit the same bottleneck in the network. This simple "locality" based approximation is unlikely to address the need to differentiate between flows that are not sharing paths but have the same destination.

Harfoush, Bestavros, and Byers [12] use packet-pair probing for determining whether two flows originating from the same source share a bottleneck. Their technique is based on correlating end-to-end packet loss measurements to identify flows that share "similar network conditions". The main disadvantage of their technique, in addition to being dependent on packet level probing, is the requirement of cooperating senders. Rubenstein, Kurose, and Towsley [13] have developed an end-to-end technique based on packet loss or delay observations to infer whether two flows are experiencing congestion on a common set of network resources. Their methodology is based on the observation that losses or delays experienced by two packets passing through the same bottleneck exhibit some degree of positive correlation. A major shortcoming of this approach is the huge computational cost to correlate packet level measurements. Moreover, the technique assumes that the flows share a common endpoint; i.e., either the sources or the destinations of packets are co-located and collaborating, which has limited applicability.

Katabi, Bazzi, and Yang [14] have developed iterative clustering techniques that minimize entropy-based cost functions to cluster flows that share a bottleneck into groups. Their method is based on the observation that correct clustering minimizes the entropy of inter-packet spacing within clusters with an empirical distribution measured by an observer. The main advantage of their method is that it does not require sending probe traffic into the network or require cooperating senders; i.e. it is passive. However, they also indicate that their technique is robust only when the observer can monitor a large fraction of the traffic from the bottleneck link, and hence is not practical when the observer is an end-receiver.

C. Summary of Contributions and Outline

First, we summarize the contributions of this paper.

1) We propose a methodology for inferring path sharing among classes of flows based on *flow level* measurements. This method treats the "network" as a "black box" and only uses data available at a measurement point, e.g. ingress/egress, to support inference.
2) We validate and refine our methodology on actual flow level records collected at an Internet border router showing the viability of this approach. We experimentally establish a criterion for filtering out flows based on their sojourn in the system to provide better inferences.
3) Finally, we discuss possible applications of this technique which include network monitoring and root cause analysis of poor performance.

The outline of the paper is as follows. In Section II, we describe factor analysis that is used to study the throughput correlations in TCP traffic sharing paths or bottlenecks. In Section III, we show the effectiveness of this approach on actual TCP flow records collected at the border router of The University of Texas at Austin. Section IV concludes the paper.

II. Methods

The collection of flows in the network is denoted by a set $\mathcal{F}$. The size, start time (time of arrival of the first packet in a flow), end time (time of arrival of the last packet in a flow), and duration of a flow will be denoted by v_f, s_f, e_f, and $d_f = e_f - s_f$, respectively. Each flow $f \in \mathcal{F}$ belongs to a flow class $c \in \mathcal{C}$. The function $\phi : \mathcal{F} \to \mathcal{C}$ determines the class of a particular flow. We let $\mathcal{F}_c(t) = \{f \in \mathcal{F} : \phi(f) = c \text{ and } s_f \leq t < e_f\}$ denote the set of flows that belong to class c and are *active* at time t.

The *perceived throughput* for a flow is given by $r_f = v_f/d_f$[1], and the average throughput of a flow class $c \in \mathcal{C}$

[1] For one-packet flows, the perceived throughput is not well-defined.

at the measurement point can be expressed as

$$r_c(t) = \begin{cases} \frac{1}{|\mathcal{F}_c(t)|}\sum_{f\in\mathcal{F}_c(t)} r_f, & \text{if } |\mathcal{F}_c(t)| > 0,\\ 0, & \text{otherwise.} \end{cases}$$

We will estimate correlations among flow class throughputs by using temporal throughput observations at times when all of the classes are active. The condition that requires all flow classes to be active at a given time ensures that there is a class throughput observation at that time for each class under consideration[2]. For p flow classes, let the random vector $\mathbf{R} = (R_{c_1}, R_{c_2}, \ldots, R_{c_p})^T$, whose joint pdf is given by $f_{\mathbf{R}}$, denote the typical class throughputs seen by classes at a typical time. Note that it is possible that $R_{c_i} = 0$ for any $i = 1, \ldots, p$; i.e., no flow from class c_i is active at a typical time. Next, let $\mathcal{E} = \{R_{c_i} > 0, \forall c_i \in \mathcal{C}\}$, and define a random vector of throughputs conditioned on $\mathcal{E}$, $\mathbf{R}^* = (R^*_{c_1}, R^*_{c_2}, \ldots, R^*_{c_p})^T$, with a joint pdf $f_{\mathbf{R}|\mathcal{E}}$. Denote the intervals over which all flow classes are active over a measurement period T by $\mathcal{O}(T)$ (see Fig. 1). We will assume for now that $r_{c_i}(t)$ and $r_{c_j}(t)$ are realizations of ergodic random processes of throughputs of flow classes c_i and c_j, respectively. The *conditional* mean and variance of throughput for flow class c_i are defined as

$$\mu_{c_i} = \mathbb{E}[R^*_{c_i}] = \lim_{T\to\infty} \frac{1}{|\mathcal{O}(T)|} \int\limits_{\mathcal{O}(T)} r_{c_i}(t)dt, \tag{1}$$

$$\begin{aligned} \sigma^2_{c_i} &= \mathbb{E}[(R^*_{c_i} - \mu_{c_i})^2] \\ &= \lim_{T\to\infty} \frac{1}{|\mathcal{O}(T)|} \int\limits_{\mathcal{O}(T)} (r_{c_i}(t) - \mu_{c_i})^2 dt, \end{aligned} \tag{2}$$

and the *conditional* correlation of throughputs of flow classes c_i and c_j is defined using (1) and (2):

$$\begin{aligned} \rho_{c_i c_j} &= \frac{\mathbb{E}[(R^*_{c_i} - \mu_{c_i})(R^*_{c_j} - \mu_{c_j})]}{\sigma_{c_i}\sigma_{c_j}} \\ &= \lim_{T\to\infty} \frac{1}{|\mathcal{O}(T)|\sigma_{c_i}\sigma_{c_j}} \\ &\quad \times \int\limits_{\mathcal{O}(T)} (r_{c_i}(t) - \mu_{c_i})(r_{c_j}(t) - \mu_{c_j})dt. \end{aligned} \tag{3}$$

A. Factor Analysis of the Conditional Correlation Matrix

For p flow classes, we express the random vector of conditional class throughputs as $\mathbf{R}^* = (R^*_{c_1}, R^*_{c_2}, \ldots, R^*_{c_p})^T$ with a mean vector $\boldsymbol{\mu}_{\mathbf{R}^*} = (\mu_{c_1}, \mu_{c_2}, \ldots, \mu_{c_p})^T$. The idea underlying factor analysis is to consider a representation for $\mathbf{R}^*$ in terms of a random vector of m ($m \le p$) *common* factors $\mathbf{F} = (F_1, F_2, \ldots, F_m)^T$, and a random vector of *unique* factors $\mathbf{U} = (U_{c_1}, U_{c_2}, \ldots, U_{c_p})^T$. We assume that $\mathbf{R}^*$ can be expressed as

$$\mathbf{R}^* - \boldsymbol{\mu}_{\mathbf{R}^*} = \boldsymbol{\Lambda}\mathbf{F} + \mathbf{U}, \tag{4}$$

where $\boldsymbol{\Lambda}$ denotes a deterministic $p \times m$ *loading matrix*. The following additional assumptions are usually made: $\mathbb{E}[\mathbf{F}] = \mathbf{0}$, $\text{Cov}(\mathbf{F}) = \mathbb{E}[\mathbf{F}\mathbf{F}^T] = \mathbf{I}$ (orthogonal factors), $\mathbb{E}[\mathbf{U}] = \mathbf{0}$, $\text{Cov}(\mathbf{U}) = \mathbb{E}[\mathbf{U}\mathbf{U}^T] = \boldsymbol{\Psi} = \text{diag}(\psi_1, \ldots, \psi_p)$ (all covariances among class throughputs are accounted by the factors), and $\text{Cov}(\mathbf{U}, \mathbf{F}) = \mathbf{0}$. Using (4), one can write

$$\text{Cov}(\mathbf{R}^*) = \mathbb{E}[(\mathbf{R}^* - \boldsymbol{\mu}_{\mathbf{R}^*})(\mathbf{R}^* - \boldsymbol{\mu}_{\mathbf{R}^*})^T] = \boldsymbol{\Lambda}\boldsymbol{\Lambda}^T + \boldsymbol{\Psi}.$$

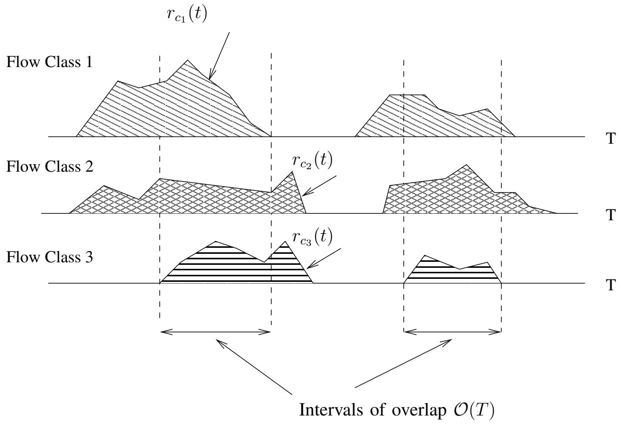

Fig. 1. Temporal overlaps of flow classes.

Alternatively, one can obtain a conditional correlation matrix $\boldsymbol{\rho}$ using (3), and express it as

$$\boldsymbol{\rho} = (\rho_{c_i c_j}) = \boldsymbol{\Lambda}\boldsymbol{\Lambda}^T + \boldsymbol{\Psi}, \tag{5}$$

where $\rho_{c_i c_i} = 1$. For factor analysis, we will use the conditional correlation matrix instead of the conditional covariance matrix, because the magnitudes of flow class throughputs can vary greatly and normalizing such measurements is preferable.

The elements of the loading matrix $\boldsymbol{\Lambda}$, Λ_{ij}, capture the degree of correlation exhibited between a given factor and variable. Estimates $\hat{\boldsymbol{\Lambda}}$ and $\hat{\boldsymbol{\Psi}}$ for $\boldsymbol{\Lambda}$ and $\boldsymbol{\Psi}$ can be determined by using the *principal component* method as follows (see [15] for more details). First, the (positive definite) conditional correlation matrix in (5) is expressed as

$$\boldsymbol{\rho} = e_1 \boldsymbol{\xi}_1 \boldsymbol{\xi}_1^T + e_2 \boldsymbol{\xi}_2 \boldsymbol{\xi}_2^T + \ldots + e_p \boldsymbol{\xi}_p \boldsymbol{\xi}_p^T,$$

where $(e_i, \boldsymbol{\xi}_i)$ are the eigenvalue-eigenvector pairs such that $e_1 \ge e_2 \ge \ldots \ge e_p > 0$. $\boldsymbol{\Lambda}$ and $\boldsymbol{\Psi}$ can be determined by taking the largest m eigenvalues, and by approximating $\boldsymbol{\rho}$ as

$$\begin{aligned} \boldsymbol{\rho} &\approx \hat{\boldsymbol{\Lambda}}\hat{\boldsymbol{\Lambda}}^T + \hat{\boldsymbol{\Psi}} \\ &= (\sqrt{e_1}\boldsymbol{\xi}_1, \ldots, \sqrt{e_m}\boldsymbol{\xi}_m) \\ &\quad \times (\sqrt{e_1}\boldsymbol{\xi}_1, \ldots, \sqrt{e_m}\boldsymbol{\xi}_m)^T \\ &\quad + \begin{pmatrix} \hat{\psi}_1 & 0 & \cdots & 0 \\ 0 & \hat{\psi}_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \hat{\psi}_p \end{pmatrix}, \end{aligned} \tag{6}$$

so that $\hat{\Lambda}^2_{i1} + \hat{\Lambda}^2_{i2} + \ldots + \hat{\Lambda}^2_{im)} + \hat{\psi}_i = h_i^2 + \hat{\psi}_i = 1$, where h_i^2 is called the *communality*, and $\hat{\psi}_i$ is called the

[2] Note that having all flow classes active at a given time is a stringent requirement. However, we choose to impose our stringent condition to guarantee positive definiteness of the throughput correlation matrix that is used in this section.

specific "variance". The communality represents the portion of the normalized variance of $R^*_{c_i}$ that is accounted by the m common factors, while $\hat{\psi}_i$ reflects the portion of the normalized variance due to a factor that is unique to $R^*_{c_i}$.

When using the principal component method to "factor" the correlation matrix *without any assumptions on the distribution of variables*, there are only ad hoc heuristics for determining the sufficiency of the number of factors m in the model. In exploratory studies, one common approach to determine m is to use Kaiser's rule [15]. Kaiser's rule proposes selecting factors whose normalized variances (given by e_j) are greater than 1. The intuition behind this rule is that a factor that has a variance less than 1 contains less information than a normalized original variable does. The number of factors m used in the model needs to account for a "reasonable" proportion of variance. The proportion of the total variance due to the jth factor is given by e_j/p, i.e., the jth eigenvalue divided by the number of classes. If the proportion of total variance captured by the common factors is "high", then we say that the factors have a strong explanatory power.

B. Interpretation of Factor Loadings

For our application, the factors represent shared sources of variation in class throughputs which are most likely due to shared congested resources. We examine the *magnitude* of factor loadings across a row (corresponding to flow classes) of $\mathbf{\Lambda}$ and mark the one(s) with the largest magnitude(s). The columns that are marked indicate the factors that are contributing the most to the variations in the throughput for that class. We identify the flow classes that share a factor as classes likely to share a bottleneck.

Note that the loading matrix is determined only up to an orthogonal rotation matrix $\mathbf{\Gamma}$. If $\mathbf{\Lambda}^* = \mathbf{\Lambda}\mathbf{\Gamma}$, then

$$\boldsymbol{\rho} = \mathbf{\Lambda}^*\mathbf{\Lambda}^{*T} + \mathbf{\Psi} = \mathbf{\Lambda}\mathbf{\Gamma}\mathbf{\Gamma}^T\mathbf{\Lambda}^T + \mathbf{\Psi} = \mathbf{\Lambda}\mathbf{\Lambda}^T + \mathbf{\Psi}.$$

Throughout this paper, we rotate the loading matrix to obtain a better description of the factors by using a common method in factor analysis called *varimax* rotation [15]. Varimax rotation attempts to find a rotation matrix $\mathbf{\Gamma}$ such that the squares of the loadings on each factor are as spread out as possible. Hence, the interpretation of loadings is easier with $\mathbf{\Lambda}^*$.

III. Analysis of TCP Flow Records

A. Dataset

We used flow records collected at the border router of The University of Texas at Austin on November 6, 2002, between 12:58 PM and 2:07 PM CST. The dataset consisted of 5,173,385 TCP flow records out of a total of 5,866,602 flow records. The records contained both the incoming and outgoing traffic from UT Austin. The IP addresses belonging to UT Austin were made anonymous to protect privacy.

B. Preprocessing

Flow throughput is not defined for one-packet flows, because the time between the first and the last packet is zero. More importantly, flows with short durations may not have an opportunity to "learn" the congestion state of the network. In addition, the throughput of a flow that lasts a very long time may not reveal the variations in the network's congestion state during the flow's sojourn. Hence, we filter out all flow records whose durations are shorter than one threshold or longer than another threshold in order to capture the correlations among flow class throughputs. We will experimentally establish that the mean duration of the flows under consideration can serve as the lower threshold for omitting flow records in order to avoid the "noise" introduced by short flows into flow class throughputs.

Next, we apply our methodology to two example cases to infer path or bottleneck sharing based on flow records. These examples are used to assess the effectiveness of our "black box" approach, and thus, are chosen so that we have some idea about possible path sharing among considered classes (for example, by using the `traceroute` utility). We use the `nslookup` and `whois` utilities to associate IP addresses with ISPs and CPs. In the first case, we analyze a set of TCP flow classes that originate from three different ISPs in the United Kingdom, and identify which classes originate from the same ISP. In the second case, we identify Web traffic either originating from the same CP or sharing a bottleneck when traversing the Internet.

1) Case 1: We defined five flow classes with source IP addresses belonging to three different ISPs in the United Kingdom: One class for BBC Internet Services, two classes for British Telecom, and two classes for PIPEX. There is reasonable certainty that flow classes that belong to an ISP share infrastructure. We chose to analyze intercontinental traffic, because the flow classes were likely to experience congestion on relatively slow intercontinental links, although perhaps not the same ones. As a validation for the factor analysis results to be considered next, we used `traceroute` to find the route that packets followed from UT Austin to each of these classes' destinations and assumed that the traffic from these ISPs to UT Austin traversed the same set of networking infrastructure reported by `traceroute`. We inspected `traceroute` results to conclude that these ISPs did not share the same intercontinental links.

2) Case 2: We selected five different CP's Web traffic from the flow records: MSN, HotMail, CNN, Yahoo!, and Google. We assumed that Web traffic belonging to these CPs had potentially experienced congestion due to high demand for content. Again, we used `traceroute` to find the route that packets followed from UT Austin to each of these destinations and assumed that the traffic from these CPs to UT Austin traversed the same set of networking infrastructure reported by `traceroute`. Web addresses containing MSN and HotMail were traced to the Microsoft Corporation, Redmond, Washington. Web addresses containing CNN were traced to Atlanta, Georgia. A fraction of Web traffic from Yahoo! and Google appeared to visit a router in Santa Clara, California. We selected the source addresses belonging to Yahoo! and Google that appeared to traverse common infrastructure in Santa Clara. We accordingly defined five flow classes: MSN,

TABLE I

FACTOR ANALYSIS RESULTS: FACTOR LOADINGS, COMMUNALITIES, AND PERCENTAGE OF VARIANCE CAPTURED BY THE FACTORS ARE SHOWN. CLASS 1: BBC, CLASS 2: BRITISH TELECOM 1, CLASS 3: BRITISH TELECOM 2, CLASS 4: PIPEX 1, CLASS 5: PIPEX 2.

-	Factor 1	Factor 2	Factor 3	Communality
Class 1	-0.049	0.035	$\boxed{0.976}$	0.955
Class 2	0.186	$\boxed{0.839}$	0.009	0.738
Class 3	-0.305	$\boxed{0.772}$	0.021	0.690
Class 4	$\boxed{0.821}$	0.103	-0.327	0.792
Class 5	$\boxed{0.883}$	-0.157	0.166	0.832
% Var	0.317	0.267	0.217	0.802

TABLE II

FACTOR ANALYSIS RESULTS: FACTOR LOADINGS, COMMUNALITIES, AND PERCENTAGE OF VARIANCE CAPTURED BY THE FACTORS ARE SHOWN. CLASS 1: MSN, CLASS 2: HOTMAIL, CLASS 3: CNN, CLASS 4: YAHOO! (SANTA CLARA), CLASS 5: GOOGLE (SANTA CLARA).

-	Factor 1	Factor 2	Factor 3	Communality
Class 1	-0.420	$\boxed{0.666}$	-0.358	0.749
Class 2	-0.244	$\boxed{-0.880}$	-0.091	0.843
Class 3	0.062	0.044	$\boxed{-0.917}$	0.846
Class 4	$\boxed{0.821}$	0.068	0.054	0.682
Class 5	$\boxed{0.687}$	-0.003	-0.470	0.694
% Var	0.277	0.245	0.240	0.763

HotMail, CNN, Yahoo! (Santa Clara), and Google (Santa Clara).

C. Validation of Methodology

We assume that over 1-hour periods, class throughputs can be modelled as stationary processes. Furthermore, we assume that the packets from a given TCP flow follow the same route[3]. Such assumptions, although idealized, are not completely unrealistic for our 1-hour long flow measurements.

Recall that in Section III-B, we have explained the need for filtering out TCP flow records to obtain better inferences on path sharing. For the first case described above, we filtered out all flow records whose durations were less than 75 seconds (approximately, the mean duration of flows under consideration) or greater than 150 seconds. We constructed the sample conditional correlation matrix (to estimate (5)) by discretizing the measurement period into 1 second intervals for our two cases. We assumed that the throughput of a flow at a discretized interval was equal to its "continuous-time" throughput if the flow was active anytime during that interval. We used Kaiser's rule [15] described in Section II-A to determine the number of factors. We performed factor analysis on the matrix by using (6) to obtain $\hat{\mathbf{\Lambda}}$ and $\hat{\mathbf{\Psi}}$. In each case, varimax rotation was applied to the determined loading matrix. The results of factor analysis with three factors on the throughput correlation matrix are given in Table I. For the second case, we filtered out all flow records whose durations were less than 19 seconds (approximately, the mean duration of flows under consideration) or greater than 150 seconds. The results of factor analysis with three factors (determined by Kaiser's rule) on the throughput correlation matrix are given in Table II. For each case, the accuracy of the proportion of total variance captured by the common factors was determined by using confidence intervals. Since the distribution of data was unknown, we resorted to the *bootstrap method* [17] to compute 95% bias-corrected and accelerated (BC$_a$) confidence intervals for the proportions of total variance captured by the common factors and reported in Tables I and II. The computed intervals were [0.794, 0.809] and [0.741, 0.781], respectively. Therefore, we concluded that three factors, which accounted for more than 70% of the total variance in data, could sufficiently model variations in five class throughputs, and the loadings that were marked successfully identify the flow classes that share infrastructure in both cases. The inferences for path sharing are consistent with those obtained by using `traceroute`.

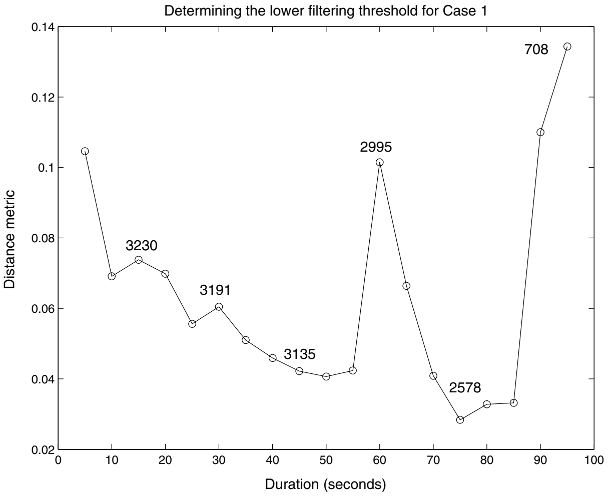

Fig. 2. Determination of the lower threshold for filtering flow records in Case 1. The mean duration of flows under consideration was 70.4 seconds. Numbers of discretized overlaps used in computing some of the temporal correlations are also shown.

Note, however, that if we filter out too many records, the number of samples available for statistical analysis will decrease, reducing our ability to infer path sharing. In order to assess the choice for the lower filtering threshold (approximately, the mean duration of flows under consideration), we introduce a "distance metric" which gives us a measure of the ability of the factors to capture path sharing among flow classes:

$$d := \frac{1}{p \times m} \sum_{i=1}^{p} \sum_{j=1}^{m} (\Lambda_{ij}^{0} - |\hat{\Lambda}_{ij}|)^2, \qquad (7)$$

where, based on `traceroute` results from Sections III-B.1 and III-B.2, we set $\Lambda_{ij}^{0} = 1$ if the flow class c_i shares the

[3] This assumption is supported by the empirical measurements in [16].

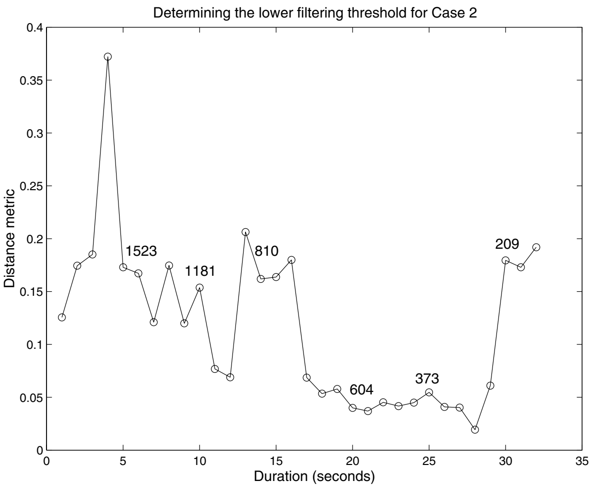

Fig. 3. Determination of the lower threshold for filtering flow records in Case 2. The mean duration of flows under consideration was 17.5 seconds. Numbers of discretized overlaps used in computing some of the temporal correlations are also shown.

factor j, and set $\Lambda^0_{ij} = 0$ otherwise, which corresponds to ideal loadings. Figs. 2 and 3 show the distance metric given in (7) versus the lower threshold for filtering out flow records in the two cases we consider in this section. The plots illustrate that the distance metric is low (corresponding to the ability of the factors to capture path sharing) if flow records with durations less than the sample mean are filtered out. Note that as the filtering threshold is increased beyond the mean, the ability of the factors used to infer path sharing decreases since the number of samples used for statistical analysis decreases. Figs. 2 and 3 also show the numbers of discretized overlaps used in computing some of the temporal correlations.

The potential power of our technique in root cause analysis may be illustrated by considering the results in Table II. For example, suppose that the users of services from MSN and HotMail at UT Austin were experiencing poor performance (delay), and UT Austin's network managers were capable of verifying that utilization of the local network was low. Treating the outside network as a "black box" (i.e., no knowledge about the utilization levels of access links or routing information of outside network), network managers could infer that poor performance was not due to the access links connecting UT Austin to the Internet, because the flow classes did not have one common factor that would indicate a bottleneck shared by all classes. The network managers could then hypothesize that the cause for poor performance was either at the CP's server or a bottleneck link visited by both flow classes in the Internet.

IV. Conclusion

Our proposed approach for inferring path sharing based on flow records can serve as a tool for network monitoring and root cause analysis of poor performance. The methodology may also be used to assess an ISP's routing diversity, hence robustness, upon inferring that none of the flow classes share a congested resource.

One drawback of our approach is that when the offered loads are low, the amount of temporal overlaps among flow classes will be low. Thus, to collect an adequate sample for statistical analysis, records will need to be collected over an extended period of time. Unfortunately, over longer measurement periods, the traffic/network may change making inferences based on flow records difficult. We are currently in the process of developing more extensive validation scenarios, statistical tests, and employing confidence intervals to assess our proposed methods. We are also considering to apply factor analysis on a correlation matrix whose elements are estimated by using pairwise class throughput correlations in order to mitigate the inadequate sample size problem that arises when loads offered by flow classes are low.

Acknowledgments

The authors would like to thank Mr. Jason Wang and Prof. William C. Bard for providing the flow records collected at UT Austin's border router.

References

[1] L. Massoulié and J. W. Roberts, "Bandwidth sharing and admission control for elastic traffic," *Telecommunication Systems*, vol. 15, pp. 185–201, June 2000.

[2] G. de Veciana, T.-J. Lee, and T. P. Konstantopoulos, "Stability and performance analysis of networks supporting elastic services," *IEEE/ACM Trans. on Networking*, vol. 9, no. 1, pp. 2–14, Feb. 2001.

[3] S. B. Fred, T. Bonald, A. Proutiere, G. Régnié, and J. W. Roberts, "Statistical bandwidth sharing: a study of congestion at flow level," in *Proc. ACM Conf. on Applications, Tech., Arch., and Protocols for Computer Communications*, Aug. 2001, pp. 111–122.

[4] *IP Monitoring Project*. http://ipmon.sprintlabs.com: Sprint Corporation.

[5] *NetFlow*. http://www.cisco.com: Cisco Systems, Inc.

[6] *Argus*. http://www.qosient.com/argus: QoSient, llc.

[7] *sFlow*. http://www.sflow.org: sFlow.org.

[8] C. Estan and G. Varghese, "New directions in traffic measurement and accounting," in *Proc. ACM SIGCOMM Workshop on Internet Measurement*, Nov. 2001, pp. 75–80.

[9] N. Duffield, C. Lund, and M. Thorup, "Properties and prediction of flow statistics from sampled packet streams," in *Proc. ACM SIGCOMM Workshop on Internet Measurement*, Nov. 2002, pp. 159–171.

[10] A. Feldmann, A. Gilbert, P. Huang, and W. Willinger, "Dynamics of IP traffic: a study of the role of variability and the impact of control," in *Proc. ACM Conf. on Applications, Tech., Arch., and Protocols for Computer Communications*, Aug. 1999, pp. 301–313.

[11] S. Savage, N. Cardwell, and T. Anderson, "The case for informed transport protocols," in *Proc. IEEE Workshop on Hot Topics in Operating Systems*, Mar. 1999, pp. 58–63.

[12] K. Harfoush, A. Bestavros, and J. Byers, "Robust identification of shared losses using end-to-end unicast probes," in *Proc. IEEE International Conference on Network Protocols*, Nov. 2000, pp. 22–36.

[13] D. Rubenstein, J. Kurose, and D. Towsley, "Detecting shared congestion of flows via end-to-end measurement," *IEEE/ACM Trans. on Networking*, vol. 10, no. 3, pp. 381–395, June 2002.

[14] D. Katabi, I. Bazzi, and X. Yang, "A passive approach for detecting shared bottlenecks," in *Proc. IEEE International Conference on Computer Communications and Networks*, 2001, pp. 174–181.

[15] A. C. Rencher, *Multivariate Statistical Inference and Applications*. Wiley, 1998.

[16] V. Paxson, "End-to-end routing behavior in the Internet," *IEEE/ACM Trans. on Networking*, vol. 5, no. 5, pp. 601–615, Oct. 1997.

[17] D. Efron and R. J. Tibshirani, *An Introduction to the Bootstrap*. Chapman & Hall, Inc., 1993.